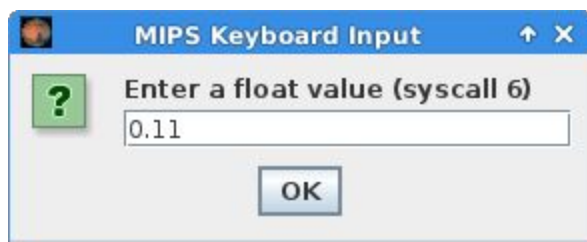


Mike Zrimsek
Computer Architecture

This program works first by asking the user to declare how many numbers (as an integer value) they are going to enter.



Once it knows how many numbers are going to be entered, a loop counter is set to 0 to keep track of how many iterations of the loop have been completed. During each loop iteration the user is prompted to enter a float value.



```
How many numbers: **** user input : 10
Enter number: **** user input : 0.11
Value of numbers entered: 0.11
Enter number: **** user input : 0.34
```

During each iteration, a register is incremented with the entered float value to keep track of the total value of the numbers entered and prints that value to the console. If the loop counter is less than the amount of numbers to be entered, the program jumps back to the top of the loop and repeats, otherwise it breaks out to begin doing the calculation of the average of the total. To calculate the average the program takes the number entered for the amount of numbers to be entered and converts it to a float so division can be done accurately.

```
Value of numbers entered: 8.77**** user input : 0.12
Value of numbers entered: 8.89**** user input : 0.87
Value of numbers entered: 9.76**** user input : 0.56
Value of numbers entered: 10.320001
Mean of numbers: 1.0320001
-- program is finished running --
```

To calculate the mean of the numbers entered, the program takes the total value that has been tracked in the loop and divides it by the amount of numbers that were entered (using the float converted value), and then subsequently prints it to the console.

This project really helped solidify in my mind how floating point values work at a low level. Being able to see how the values changed in the registers as the program ran will be helpful when it comes to having to actually work out these arithmetic operation by hand. In addition, the idea that you have to convert all operands to a float format prior to do anything kind of operation was something that confused me at first, but after having implemented some of it, makes total sense. In almost every modern language I can think of the compiler would handle that kind of stuff for you, but having to manually do that conversion in this project helped drive that point about the way floating point numbers work home.